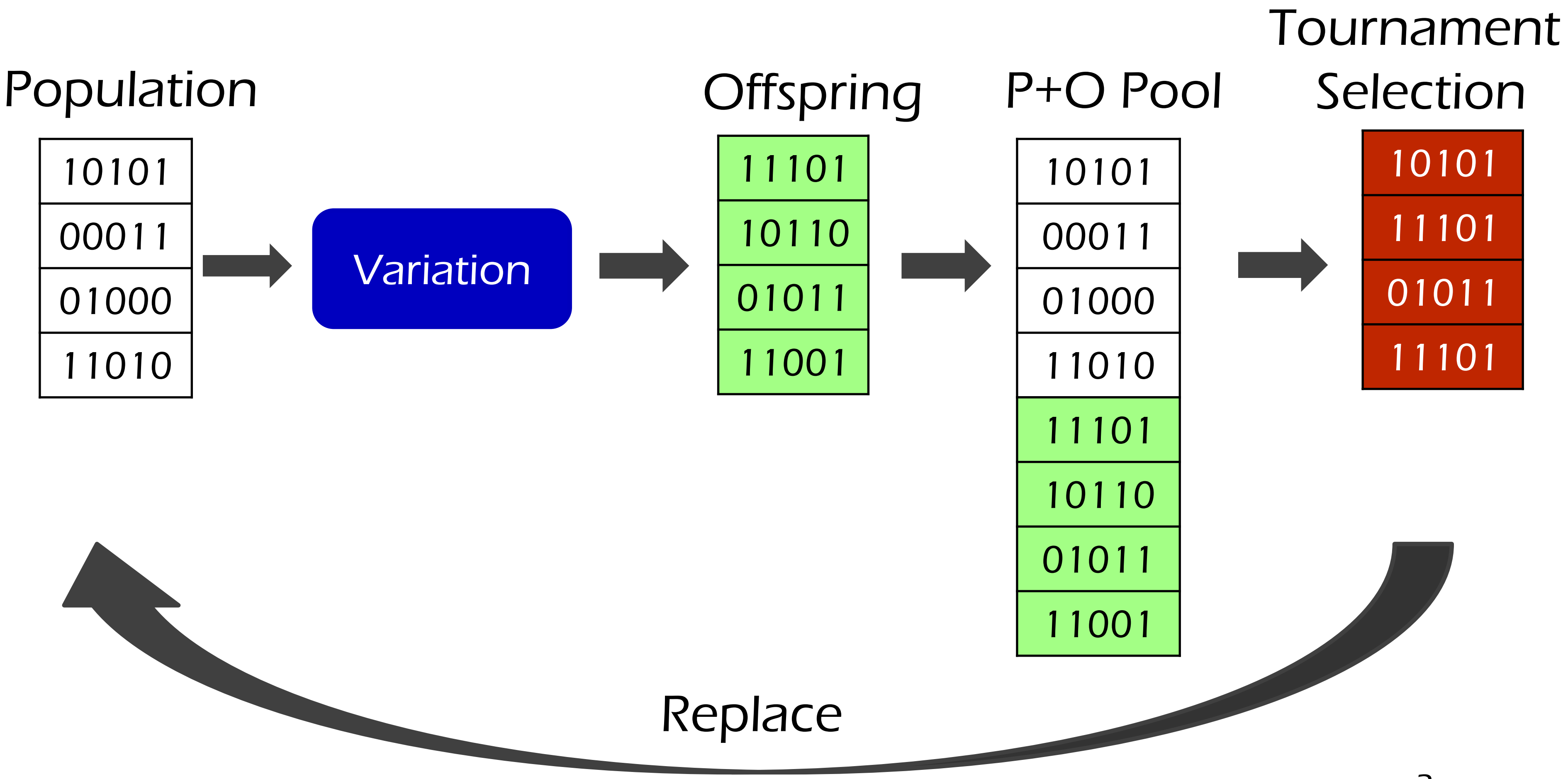
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

MODEL-BASED EVOLUTIONARY ALGORITHMS

-

CÁC THUẬT TOÁN TIẾN HÓA THEO MÔ HÌNH

Genetic Algorithm (POPOP)



Các phép biến đổi (Variation)

- Phép biến đổi: được thực hiện trên tập lựa chọn *S* (selection set) nhằm tạo ra các cá thể mới (offspring) từ các cá thể cha mẹ (parents).

- Thuật giải di truyền GA có 2 phép biến đổi chính:

➢ Lai ghép (crossover), còn được gọi là Tái tổ hợp (recombination).

➢ Đột biến (mutation)

Phép biến đổi – Lai ghép

Phép lai ghép: Tạo ra cá thể mới bằng cách kết hợp kiểu gen của những cá thể (có độ thích nghi cao) trong quần thể.

Cá thể cha mẹ (Parents)	Phép lai (Crossover)	Cá thể con (Offspring)
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Lai một điểm (One-point crossover – 1X)	0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Lai hai điểm (Two-point crossover – 2X)	0 0 0 1 1 1 1 1 1 1 1 0 0 0 1 1 1 0 0 0 0 0 0 0 0 1 1 1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Lai đồng nhất (Uniform crossover – UX)	1 0 1 1 0 0 0 1 0 1 1 0 1 0 0 1 0 0 1 1 1 0 1 0 0 1 0 1

Lai một điểm

0	1	0	0	0	1	1	0	1	0
0	0	1	0	1	1	0	1	1	1

Lai một điểm

0	1	0	0	0	1	1	0	1	0
0	0	1	0	1	1	0	1	1	1

Lai một điểm

0	1	0	0	0	1	1	0	1	0
0	0	1	0	1	1	0	1	1	1

0	1	0	0	1	1	0	1	1	1
0	0	1	0	0	1	1	0	1	0

Lai hai điểm

0	1	0	0	0	1	1	0	1	0
0	0	1	0	1	1	0	1	1	1

Lai hai điểm

0	1	0	0	0	1	1	0	1	0
0	0	1	0	1	1	0	1	1	1

Lai hai điểm

0	1	0	0	0	1	1	0	1	0
0	0	1	0	1	1	0	1	1	1

0	1	0	0	1	1	0	1	1	0
0	0	1	0	0	1	1	0	1	1

Lai đồng nhất

0	1	0	0	0	1	1	0	1	0
0	0	1	0	1	1	0	1	1	1

Lai đồng nhất

0	1	0	0	0	1	1	0	1	0
0	0	1	0	1	1	0	1	1	1

0.14	0.35	0.67	0.49	0.72	0.53	0.03	0.91	0.6	0.25

Lai đồng nhất

0	1	0	0	0	1	1	0	1	0
0	0	1	0	1	1	0	1	1	1

0.14	0.35	0.67	0.49	0.72	0.53	0.03	0.91	0.6	0.25

0	0	0	0	0	1	0	0	1	1
0	1	1	0	1	1	1	1	1	0

Phép biến đổi – Đột biến

- Phép đột biến: Tạo ra cá thể mới bằng cách phát sinh những biến đổi ngẫu nhiên trên kiểu gen của các cá thể trong quần thể.
- Xác suất đột biến thường nhỏ.

Đột biến

Ví dụ: Xác suất đột biến 0.05

0	1	0	0	0	1	1	0	1	0
0	0	1	0	1	1	0	1	1	1
1	0	0	0	1	1	0	0	1	1
0	1	1	1	0	0	0	0	1	0
0	0	1	1	1	0	0	0	0	1
1	0	0	1	1	1	1	0	0	0

0	1	0	0	0	1	1	0	1	0
0	0	1	1	1	1	0	1	1	1
1	0	0	0	1	1	0	0	1	1
0	1	1	1	0	0	0	0	0	0
0	0	0	1	1	0	0	0	0	1
1	0	0	1	1	1	1	0	0	0

Câu hỏi thảo luận

- Các thuật toán tiến hoá hoạt động hiệu quả khi nào?

- Những trường hợp nào có thể khiến các thuật toán tiến hoá gặp khó khăn?

OneMax Function

$$f(\boldsymbol{x}) = \sum_{i=1}^{l} x_i$$

f(*0**)	Giá trị
0000	0
0001	1
0010	1
0011	2
1000	1
1001	2
1010	2
1011	3
Trung bình	12/8 = 1.5

f(*1**)	Giá trị
0100	1
0101	2
0110	2
0111	3
1100	2
1101	3
1110	3
1111	4
Trung bình	20/8 = 2.5

OneMax Function

$$f(\boldsymbol{x}) = \sum_{i=1}^{l} x_i$$

f(*00*)	Giá trị
0000	0
0001	1
1000	1
1001	2
Trung bình	4/4 = 1

f(*11*)	Giá trị
0110	2
0111	3
1110	3
1111	4
Trung bình	12/4 = 3

OneMax Function

$$f(\boldsymbol{x}) = \sum_{i=1}^{l} x_i$$

f(*000)	Giá trị
0000	0
1000	1
Trung bình	1/2 = 0.5

f(*111)	Giá trị
0111	3
1111	4
Trung bình	7/2 = 3.5

OneMax Function

$$f(\boldsymbol{x}) = \sum_{i=1}^{l} x_i$$

f(0000)	Giá trị
0000	0
Trung bình	0/1 = 0

f(*111)	Giá trị
1111	4
Trung bình	4/1 = 4

Schema Theorem

Định lý lược đồ:

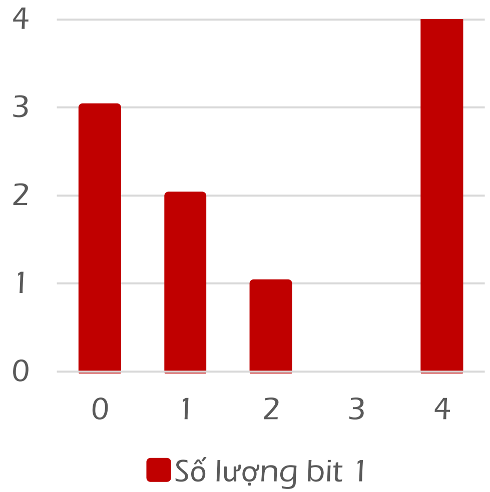
- Trong quá trình hoạt động của một thuật giải di truyền đơn giản, các lược đồ có chiều dài định nghĩa ngắn với có độ thích nghi trên trung bình sẽ nhận được số bản sao tăng luỹ thừa trong các thế hệ nối tiếp nhau.

Trap Function

$$f(\boldsymbol{x}) = f_{\text{TRAP}}(u)$$

$$u = \sum_{i=1}^{4} x_i$$

$$f_{\text{TRAP}}(u) = \begin{cases} 4 & \text{nếu u} = 4 \\ 3 - u & \text{nếu u} < 4 \end{cases}$$

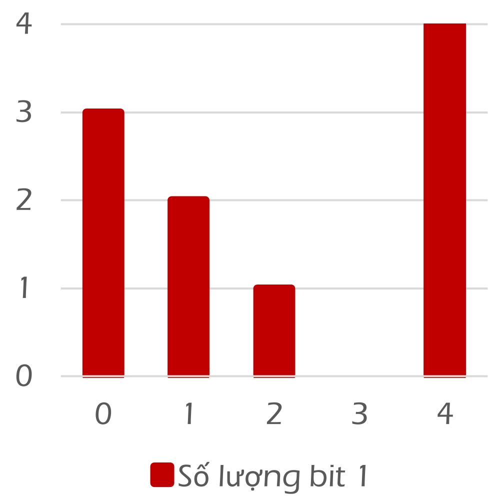


Trap Function

$$f(\boldsymbol{x}) = f_{\text{TRAP}}(u)$$

$$u = \sum_{i=1}^{4} x_i$$

$$f_{\text{TRAP}}(u) = \begin{cases} 4 & \text{nếu u} = 4 \\ 3 - u & \text{nếu u} < 4 \end{cases}$$



f(*0**)	Giá trị
0000	3
0001	2
0010	2
0011	1
1000	2
1001	1
1010	1
1011	0
Trung bình	12/8 = 1.5

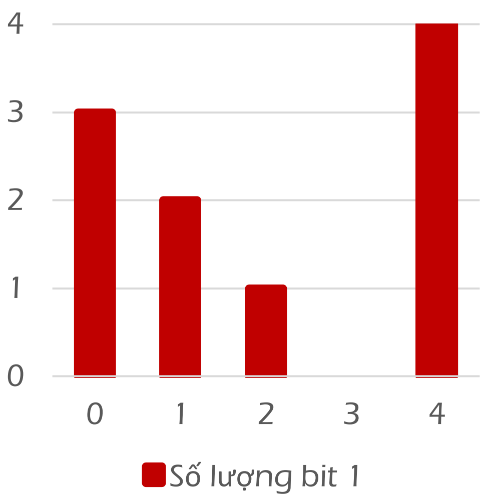
f(*1**)	Giá trị
0100	2
0101	1
0110	1
0111	0
1100	1
1101	0
1110	0
1111	4
Trung bình	9/8 = 1.125

Trap Function

$$f(\boldsymbol{x}) = f_{\mathrm{TRAP}}(u)$$

$$u = \sum_{i=1}^{4} x_i$$

$$f_{\mathrm{TRAP}}(u) = \begin{cases} 4 & \text{nếu u} = 4 \\ 3 - u & \text{nếu u} < 4 \end{cases}$$



f(*00*)	Giá trị
0000	3
0001	2
1000	2
1001	1
Trung bình	10/4 = 2.4

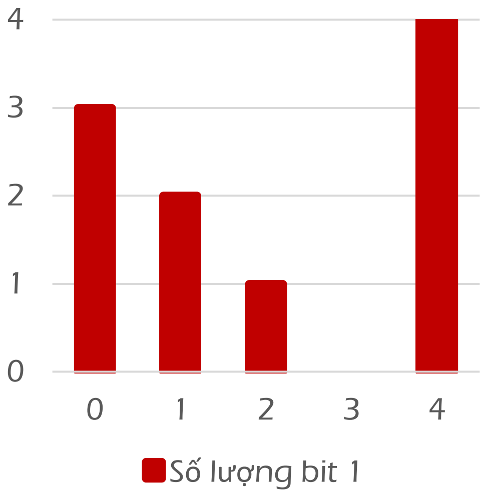
f(*11*)	Giá trị
0110	1
0111	0
1110	0
1111	4
Trung bình	5/4 = 1.25

Trap Function

$$f(\boldsymbol{x}) = f_{\mathrm{TRAP}}(u)$$

$$u = \sum_{i=1}^{4} x_i$$

$$f_{\mathrm{TRAP}}(u) = \begin{cases} 4 & \text{nếu u} = 4 \\ 3 - u & \text{nếu u} < 4 \end{cases}$$



f(*000)	Giá trị
0000	3
1000	2
Trung bình	5/2 = 2.5

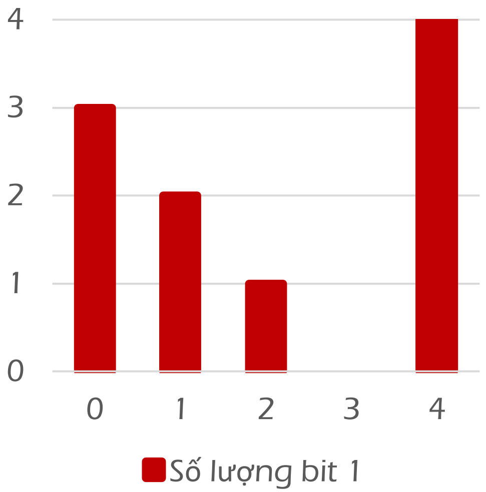
f(*111)	Giá trị
0111	0
1111	4
Trung bình	4/2 = 2

Trap Function

$$f(\boldsymbol{x}) = f_{\text{TRAP}}(u)$$

$$u = \sum_{i=1}^{4} x_i$$

$$f_{\text{TRAP}}(u) = \begin{cases} 4 & \text{nếu u} = 4 \\ 3 - u & \text{nếu u} < 4 \end{cases}$$



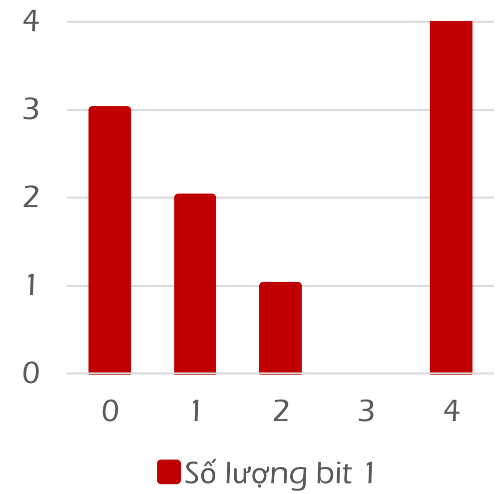
f(0000)	Giá trị
0000	3
Trung bình	3/1 = 3

f(1111)	Giá trị
1111	4
Trung bình	4/1 = 4

Concatenated Trap Function

$$f(\boldsymbol{x}) = \sum_{i=1}^{l/4} f_{\mathrm{TRAP}}\left(\sum_{j=4i-3}^{4i} x_j\right)$$

$$f_{\mathrm{TRAP}}(u) = \begin{cases} 4 & \text{nếu u} = 4 \\ 3 - u & \text{nếu u} < 4 \end{cases}$$



Ví dụ: Nối 4 Trap liên tiếp nhau.

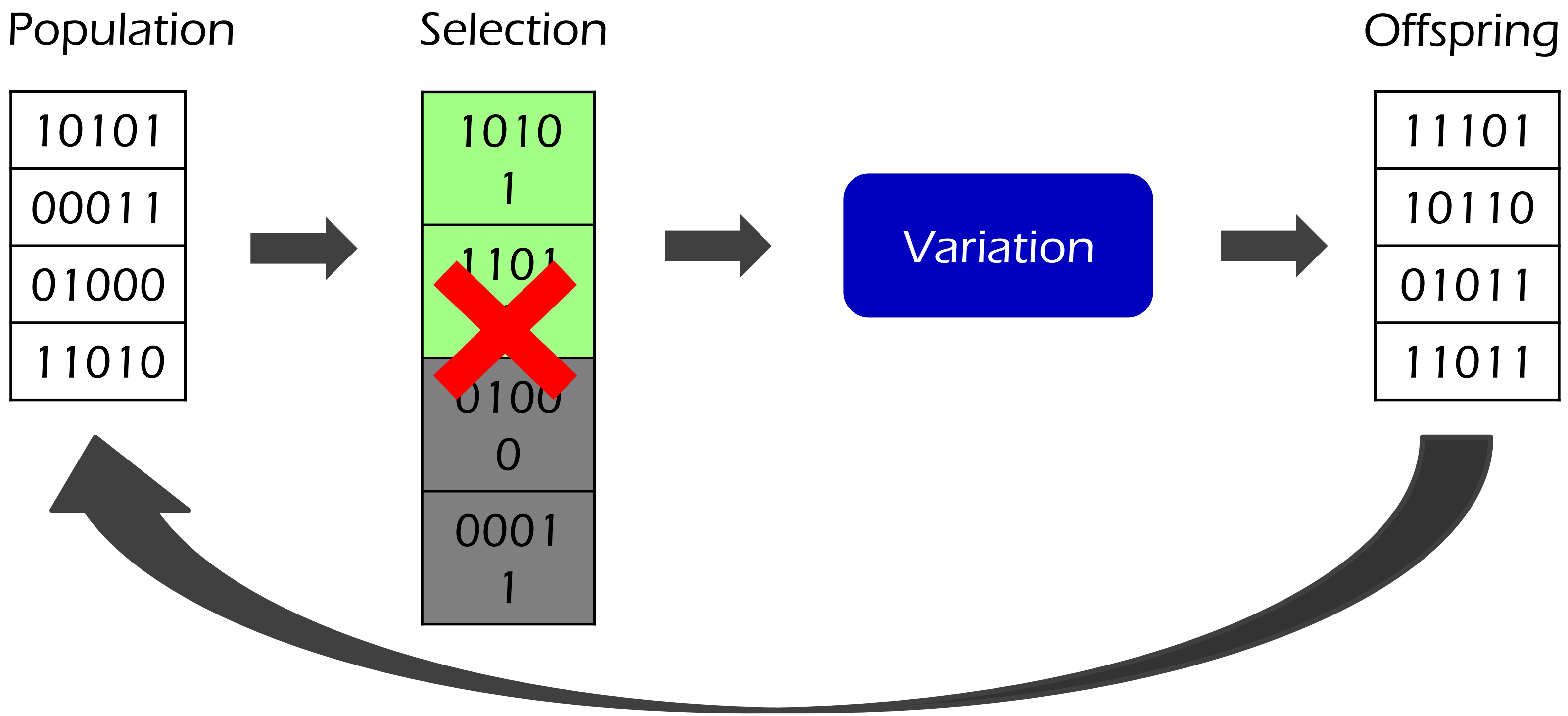
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
TRAP THỨ 1				TRAP THỨ 2				TRAP THỨ 3				TRAP THỨ 4			

Câu hỏi thảo luận

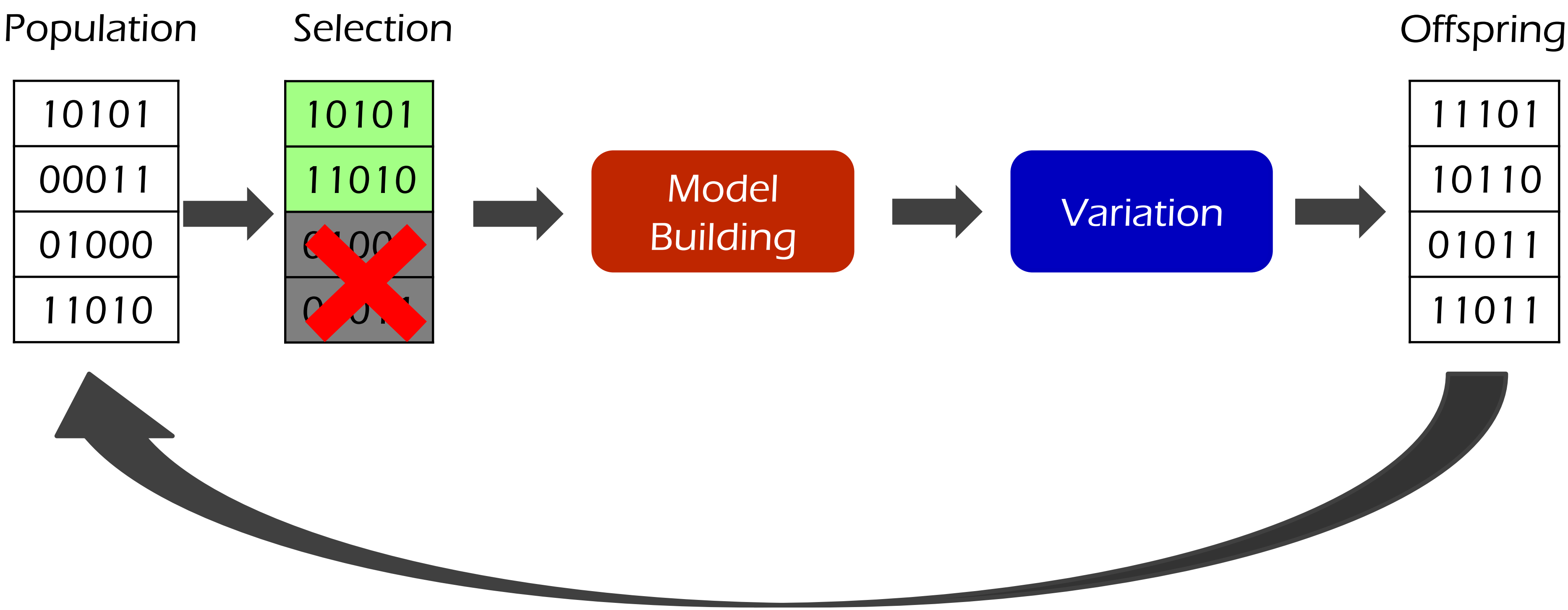
- Làm sao nhận biết cấu trúc vấn đề khi áp dụng thuật toán tiến hoá (kể cả khi giải bài toán black-box optimization)?

- Làm sao để các phép biến đổi ngẫu nhiên trở nên hiệu quả?

EA truyền thống



EA dựa trên mô hình



Các kiểu mô hình

- Mô hình đơn biến (Univariate): Xem như tất cả các biến là độc lập và không tồn tại quan hệ giữa các biến.

- Mô hình đa biến (Multivariate): Các quan hệ giữa các biến được biểu diễn.

Mô hình đơn biến (Univariate Model)

- Mô hình:

$$\{\{1\},\{2\},\{3\},\ldots,\{l-1\},\{l\}\}$$

➢ Bỏ qua tất cả quan hệ giữa các biến của bài toán.

➢ Đơn giản, không cần phải thực hiện phép học.

- Thuật giải di truyền với phép lai đồng nhất

➢ Có thể được xem như sử dụng mô hình đơn biến.

➢ Các phép biến đổi (lai ghép, đột biến) không quan tâm tới mối quan hệ (nếu tồn tại) giữa các biến.

Univariate Marginal Distribution Algorithm (UMDA)

- Mô hình:

$$\{\{1\},\{2\},\{3\},\dots,\{l-1\},\{l\}\}$$

- Biểu diễn mỗi biến là một biến ngẫu nhiên (random variable) độc lập nhau.
- Ở mỗi thế hệ, ước lượng phân phối xác suất (probability distribution) của các biến ngẫu nhiên này dựa trên quần thể (hoặc tập chọn lọc).
- Phát sinh cá thể con bằng cách lấy mẫu từ phân phối đã ước lượng.

Univariate Marginal Distribution Algorithm (UMDA)

	0	1	0	0	0	1	1	0	1	0
	0	0	1	0	1	1	0	1	1	1
	1	0	0	0	1	1	0	0	1	1
	0	1	1	1	0	0	0	0	1	0
	0	0	1	1	1	0	0	0	0	1
	1	0	0	1	1	1	1	0	0	0
P(X_i=1)	2/6	2/6	3/6	3/6	4/6	4/6	2/6	1/6	4/6	3/6

Univariate Marginal Distribution Algorithm (UMDA)

$P(X_i=1)$	2/6	2/6	3/6	3/6	4/6	4/6	2/6	1/6	4/6	3/6
r	0.12	0.57	0.03	0.72	0.61	0.87	0.09	0.34	0.65	0.23
X	1	0	1	0	1	0	1	0	1	1
r	0.54	0.14	0.32	0.96	0.82	0.04	0.75	0.01	0.25	0.64
X	?	?	?	?	?	?	?	?	?	?

Univariate Marginal Distribution Algorithm (UMDA)

- UMDA là một thuật toán tiến hoá dựa vào ước lượng phân phối (Estimation-of-Distribution Algorithm – EDA) với mô hình đơn biến (univariate model).

Mô hình đa biến (Multivariate Models)

Các mô hình đa biến có khả năng biểu diễn các mối quan hệ giữa các biến của bài toán.

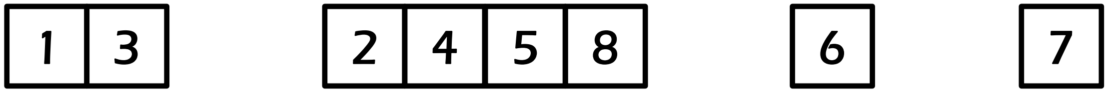
- Xây dựng mô hình dựa trên tri thức về bài toán, hoặc
- Mô hình có thể được học từ quần thể hoặc tập lựa chọn.
- Các phép biến đổi "tôn trọng" các mối quan hệ được biểu diễn bởi mô hình.

Một số loại mô hình phổ biến:

- Mô hình Marginal Product
- Mô hình Linkage Tree (Cây liên kết)
-

Marginal Product Model

- Tính chất:
  - Tập các biến được phân thành các nhóm liên kết không giao nhau (non-overlapping linkage groups).
  - Các biến trong cùng 1 nhóm liên kết được xem là có quan hệ với nhau.
  - Các biến khác nhóm liên kết được xem là không có quan hệ với nhau.
- Ví dụ:



Marginal Product Model

- Khi thực hiện phép biến đổi, các biến trong cùng 1 nhóm liên kết được biến đổi cùng lúc.
- Ví dụ:

1 3	2 4 5 8	6	7

0	1	0	0	0	1	1	0
0	0	1	0	1	1	0	1

Marginal Product Model

1	3		2	4	5	8		6		7

0	1	0	0	0	1	1	0
0	0	1	0	1	1	0	1

Marginal Product Model

1	3		2	4	5	8	6	7

0.52

0	1	0	0	0	1	1	0
0	0	1	0	1	1	0	1

1	2	3	4	5	6	7	8

Marginal Product Model

1	3

2	4	5	8

6

7

0.14

0	1	0	0	0	1	1	0
0	0	1	0	1	1	0	1

1	2	3	4	5	6	7	8

0	0	0	0	1	1	1	1
0	1	1	0	0	1	0	0

Marginal Product Model

1	3		2	4	5	8		6	7

0.76

0	1	0	0	0	1	1	0
0	0	1	0	1	1	0	1

1	2	3	4	5	6	7	8

Marginal Product Model

1	3		2	4	5	8	6	7

0.21

0	1	0	0	0	1	1	0
0	0	1	0	1	1	0	1

1	2	3	4	5	6	7	8

0	0	0	0	1	1	0	1
0	1	1	0	0	1	1	0

Marginal Product Model

- Học mô hình Marginal Product bằng cách nào?

Marginal Product Model

1. Bắt đầu từ mô hình đơn biến:

$$\{\{1\},\{2\},\{3\},\ldots\{l-1\},\{l\}\}$$

2. Tất cả nhóm liên kết được thử kết đôi với nhau:

$$\{\{1,2\},\{3\},\ldots\{l-1\},\{l\}\}$$
$$\{\{1,3\},\{2\},\ldots\{l-1\},\{l\}\}$$
$$\ldots$$
$$\{\{1\},\{2,3\},\ldots\{l-1\},\{l\}\}$$
$$\ldots$$
$$\{\{1\},\{2\},\{3\},\ldots\{l-1,l\}\}$$

3. Tính điểm **Minimum Description Length (MDL)** của từng trường hợp.
4. Kết hợp 2 nhóm liên kết tương ứng với trường hợp cho điểm MDL tốt nhất nếu điểm này tốt hơn điểm MDL của mô hình hiện tại.
5. Lặp lại từ bước 2 đến khi không thể cải thiện điểm MDL thêm nữa.

Marginal Product Model

- **Minimum Description Length (MDL)** *là gì?*

Marginal Product Model

- Mô hình tốt nhất → MDL có giá trị nhỏ nhất.
- MDL là một độ đo của độ phức tạp (complexity).

$$\mathrm{MDL} = \mathrm{CPC} + \mathrm{MC}$$

- MDL có 2 thành phần:

1. Compressed Population Complexity (CPC): Nội dung quần thể được nén bởi mô hình như thế nào?
2. Model Complexity (MC): Số lượng bit cần dùng để biểu diễn mô hình này.

Marginal Product Model

- Model Complexity: Số lượng bit cần dùng để biểu diễn mô hình.

$$\mathrm{MC} = \log_2(N+1)\sum_{I}(2^{S_I}-1)$$

- N là kích thước quần thể.
- I nhóm liên kết thứ I và có kích thước S_I (số biến trong nhóm liên kết).

Marginal Product Model

1	3		2	4	5	8		6		7

$P(X_6 = 0) = ?$
$P(X_6 = 1) = ?$

- $P(X_6 = 1)$ có N+1 khả năng: 0/N, 1/N, 2/N, ..., N/N.

Marginal Product Model

1	3	2	4	5	8	6	7

$P(X_7 = 0)$ = ?
$P(X_7 = 1)$ = ?

- $P(X_7 = 1)$ có N+1 khả năng: 0/N, 1/N, 2/N, ..., N/N.

Marginal Product Model

1	3

2	4	5	8

6

7

$P(X_1 = 0, X_3 = 0) = ?$
$P(X_1 = 0, X_3 = 1) = ?$
$P(X_1 = 1, X_3 = 0) = ?$
$P(X_1 = 1, X_3 = 1) = ?$

- $P(X_1 = 0,\ X_3 = 1)$ có N+1 khả năng: 0/N, 1/N, 2/N, ..., N/N.
- $P(X_1 = 1,\ X_3 = 0)$ có N+1 khả năng: 0/N, 1/N, 2/N, ..., N/N.
- $P(X_1 = 1,\ X_3 = 1)$ có N+1 khả năng: 0/N, 1/N, 2/N, ..., N/N.

Marginal Product Model

1	3

2	4	5	8

6

7

$P(X_2 = 0, X_4 = 0, X_5 = 0, X_8 = 0) = ?$
$P(X_2 = 0, X_4 = 0, X_5 = 0, X_8 = 1) = ?$
$P(X_2 = 0, X_4 = 0, X_5 = 1, X_8 = 0) = ?$
$P(X_2 = 0, X_4 = 0, X_5 = 1, X_8 = 1) = ?$
$P(X_2 = 0, X_4 = 1, X_5 = 0, X_8 = 0) = ?$
....
$P(X_2 = 1, X_4 = 1, X_5 = 1, X_8 = 1) = ?$

Marginal Product Model

- Compressed Population Complexity: Số lượng bit cần dùng để mô hình biểu diễn nội dung quần thể.

$$CPC = N \sum_{I} Entropy(M_I)$$

- N là kích thước quần thể.
- I nhóm liên kết thứ I và có các biến trong nhóm liên kết này có phân bố xác suất M_I.

Marginal Product Model

1	3	2	4	5	8	6	7

$P(X_6 = 0) = ?$
$P(X_6 = 1) = ?$

$$\text{Entropy}\big(P(X_6)\big)$$
$$= P(X_6 = 0).\log_2 \frac{1}{P(X_6 = 0)} + P(X_6 = 1).\log_2 \frac{1}{P(X_6 = 1)}$$

Marginal Product Model

1	3

2	4	5	8

6

7

$P(X_7 = 0) = ?$
$P(X_7 = 1) = ?$

$$\mathrm{Entropy}\big(P(X_7)\big)$$
$$= P(X_7 = 0).\log_2 \frac{1}{P(X_7 = 0)} + P(X_7 = 1).\log_2 \frac{1}{P(X_7 = 1)}$$

Marginal Product Model

1	3		2	4	5	8	6	7

$P(X_1 = 0, X_3 = 0) = ?$
$P(X_1 = 0, X_3 = 1) = ?$
$P(X_1 = 1, X_3 = 0) = ?$
$P(X_1 = 1, X_3 = 1) = ?$

$$
\begin{aligned}
&\text{Entropy}\big(P(X_1, X_3)\big) \\
&= P(X_1 = 0, X_3 = 0).\log_2 \frac{1}{P(X_1 = 0, X_3 = 0)} \\
&+ P(X_1 = 0, X_3 = 1).\log_2 \frac{1}{P(X_1 = 0, X_3 = 1)} \\
&+ P(X_1 = 1, X_3 = 0).\log_2 \frac{1}{P(X_1 = 1, X_3 = 0)} \\
&+ P(X_1 = 1, X_3 = 1).\log_2 \frac{1}{P(X_1 = 1, X_3 = 1)}
\end{aligned}
$$

Marginal Product Model

- Extended Compact Genetic Algorithm – ECGA (Harik et al., 2006)

➢ Ở mỗi thế hệ, ước lượng phân phối xác suất hợp (joint probability distribution) của các biến ngẫu nhiên trong các nhóm liên kết dựa trên quần thể (hoặc tập chọn lọc).

➢ Phát sinh cá thể con bằng cách lấy mẫu từ phân phối đã ước lượng.

Marginal Product Model

1	3

2	4	5	8

6

7

$P(X_1, X_3)$ $P(X_2, X_4, X_5, X_8)$ $P(X_6)$ $P(X_7)$

?	?	?	?	?	?	?	?
1	2	3	4	5	6	7	8

Marginal Product Model

1	3

2	4	5	8

6

7

$P(X_6 = 0) = 0.3$
$P(X_6 = 1) = 0.7$

?	?	?	?	?	?	?	?
1	2	3	4	5	6	7	8

Marginal Product Model

1	3		2	4	5	8	6	7

Probabilities

$P(X_6 = 0) = 0.3$	0.3
$P(X_6 = 1) = 0.7$	1.0

Cumulative Probabilities (Xác suất tích luỹ)

?	?	?	?	?	?	?	?

1	2	3	4	5	6	7	8

Marginal Product Model

1	3

2	4	5	8

6

7

$P(X_6 = 0) = 0.3$	0.3
$P(X_6 = 1) = 0.7$	1.0
Phát sinh ngẫu nhiên r.	

?	?	?	?	?	?	?	?

1	2	3	4	5	6	7	8

Marginal Product Model

1	3

2	4	5	8

6

7

$P(X_6 = 0) = 0.3$	0.3
$P(X_6 = 1) = 0.7$	1.0
Phát sinh ngẫu nhiên r.	
r = 0.27 → $X_6 = 0$	

?	?	?	?	?	0	?	?
1	2	3	4	5	6	7	8

Marginal Product Model

1	3

2	4	5	8

6

7

$P(X_7 = 0) = 0.6$	0.6
$P(X_7 = 1) = 0.4$	1.0
Phát sinh ngẫu nhiên r.	
r = 0.78 → $X_7 = 1$	

?	?	?	?	?	0	1	?

1	2	3	4	5	6	7	8

Marginal Product Model

1	3

2	4	5	8

6

7

$P(X_1 = 0, X_3 = 0) = 0.2$	0.2
$P(X_1 = 0, X_3 = 1) = 0.4$	0.6
$P(X_1 = 1, X_3 = 0) = 0.1$	0.7
$P(X_1 = 1, X_3 = 1) = 0.3$	1.0
Phát sinh ngẫu nhiên r.	
r = 0.64 → $X_1 = 1, X_3 = 0$	

1	?	0	?	?	0	1	?

1	2	3	4	5	6	7	8

Marginal Product Model

1	3		2	4	5	8		6		7

0000	0.04	0100	0.47	1000	0.72	1100	0.90
0001	0.16	0101	0.55	1001	0.76	1101	0.92
0010	0.25	0110	0.60	1010	0.80	1110	0.94
0011	0.30	0111	0.68	1011	0.85	1111	1.0

Phát sinh ngẫu nhiên r.

r = 0.82 → $X_2 = 1, X_4 = 0, X_5 = 1,\ X_8 = 1$

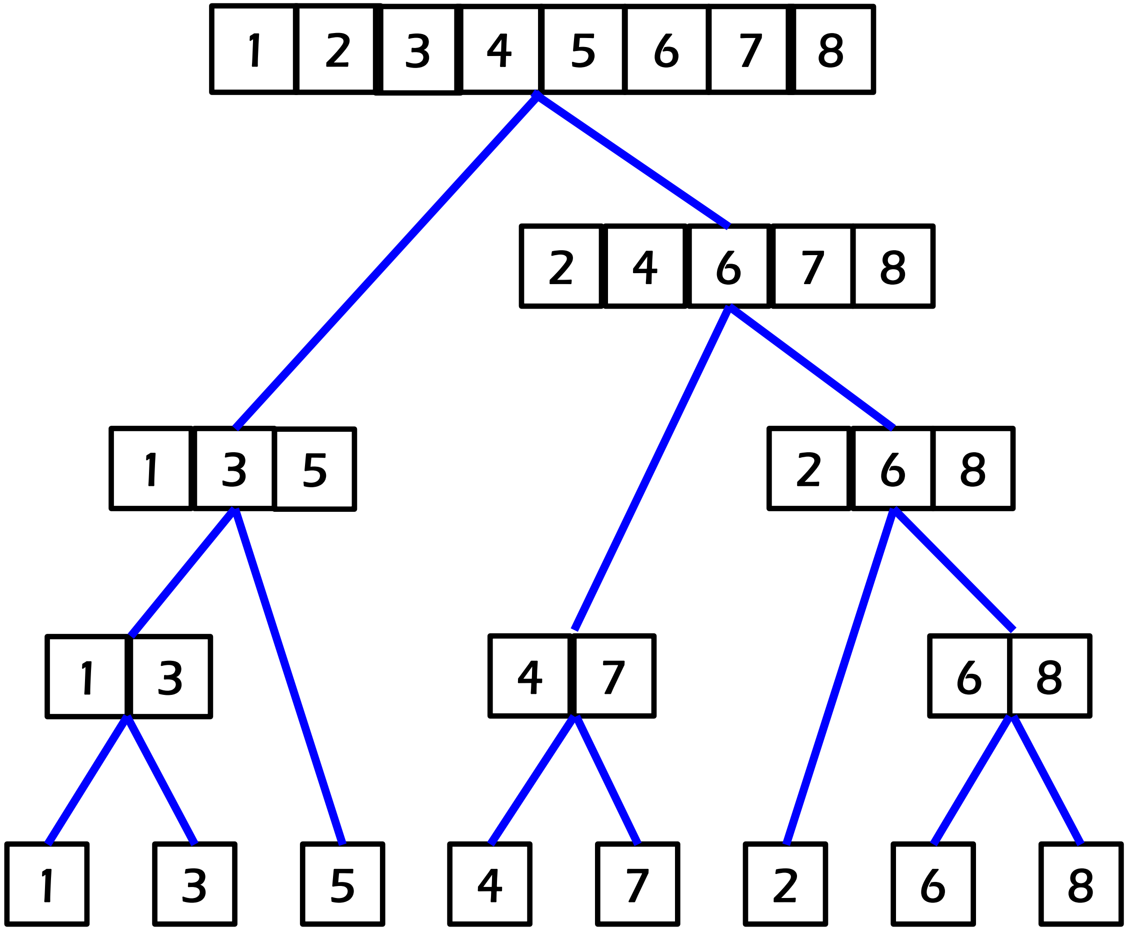
1	1	0	0	1	0	1	1

1	2	3	4	5	6	7	8

Linkage Tree Model

- Tính chất: với bài toán có l biến
  - Linkage Tree (LT) có $(2l-1)$ nút, tương ứng với $(2l-1)$ nhóm liên kết.
  - l nút lá tương ứng với l biến → các nhóm liên kết đơn biến (univariate linkage groups).
  - Các nút trung gian tương ứng với các nhóm liên kết đa biến (multivariate linkage groups).
  - Nút gốc tương ứng với 1 nhóm liên kết bao gồm tất cả l biến.

Linkage Tree Model



Linkage Tree Model

- Học mô hình Linkage Tree bằng cách nào?

1. Bắt đầu từ mô hình đơn biến.

$$\{\{1\},\{2\},\{3\},\ldots\{l-1\},\{l\}\}$$

2. Tính **Mutual Information** – MI (thông tin tương hỗ) giữa tất cả các cặp biến.
3. Sử dụng một thuật toán hierarchical clustering (phân cụm thứ bậc) để xây dựng Linkage Tree.

Linkage Tree Model

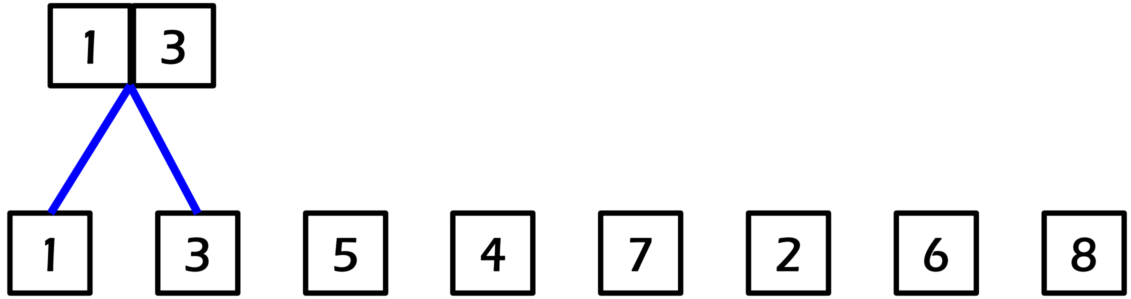
- Mutual Information (MI) là gì?

➢ MI là độ đo thể hiện mối quan hệ giữa 2 biến. Giá trị MI càng lớn thì sự phụ thuộc giữa 2 biến càng lớn.

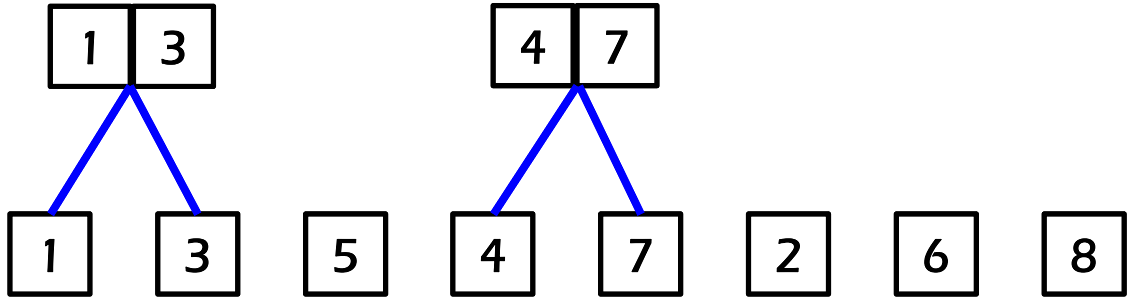
Linkage Tree Model

1 3 5 4 7 2 6 8

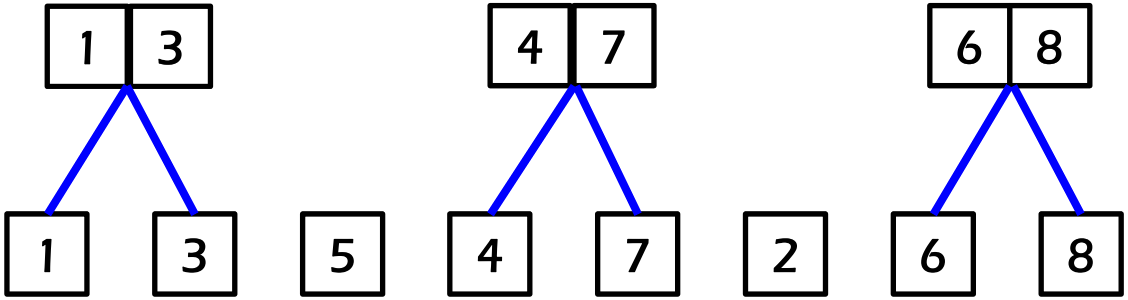
Linkage Tree Model



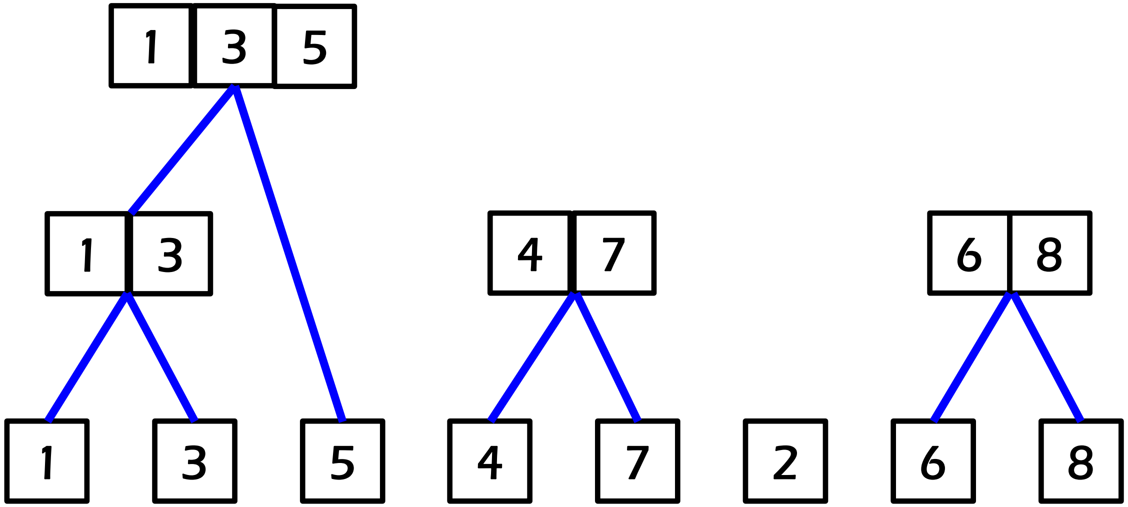
Linkage Tree Model



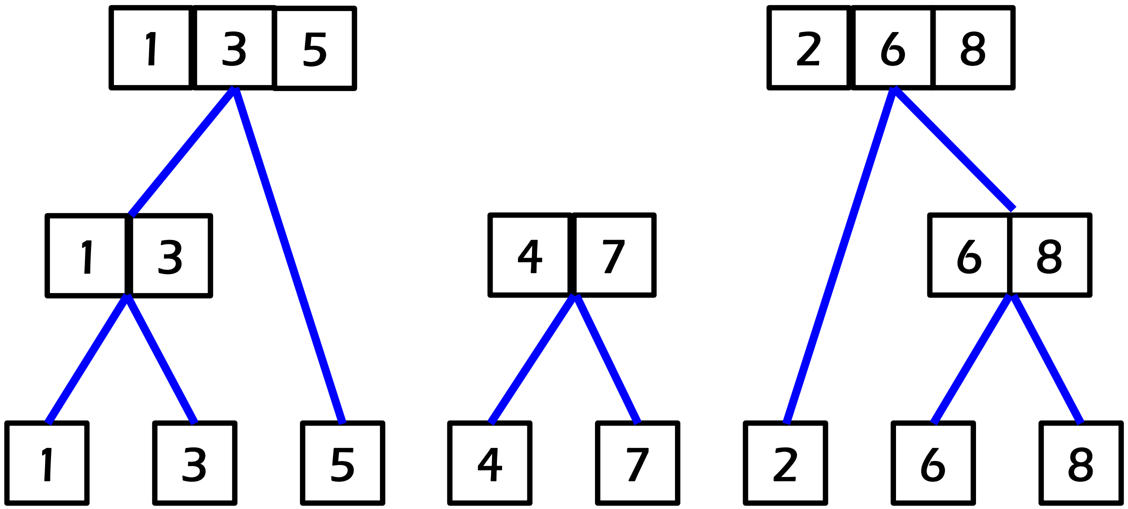
Linkage Tree Model



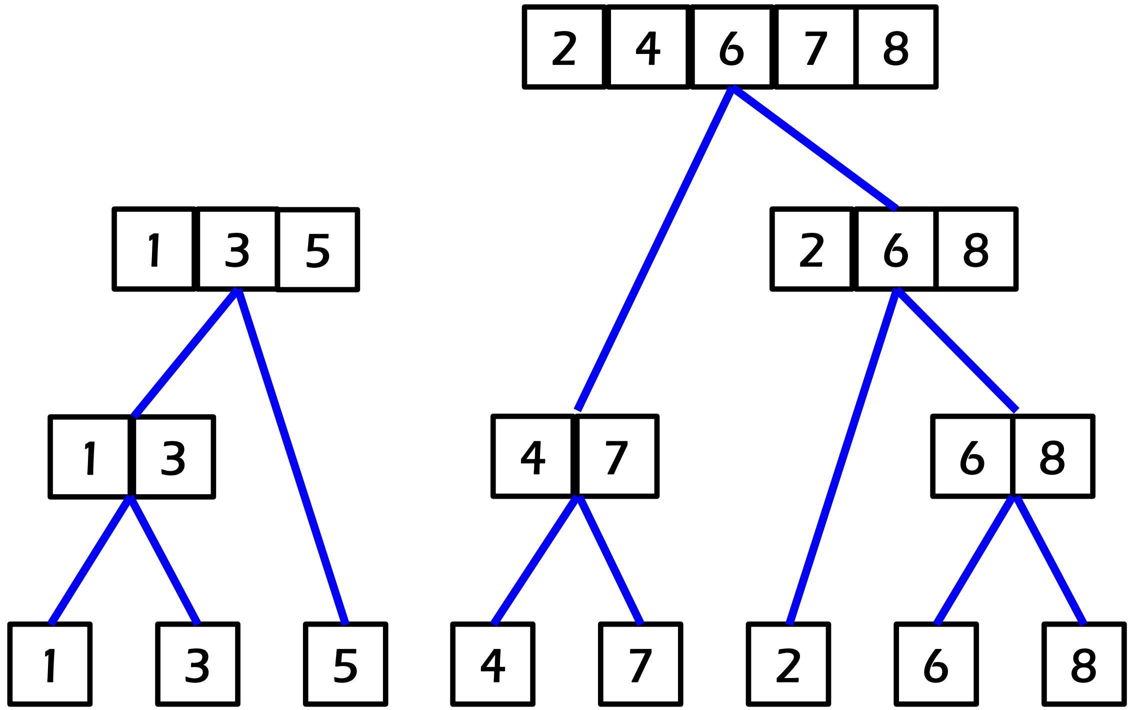
Linkage Tree Model



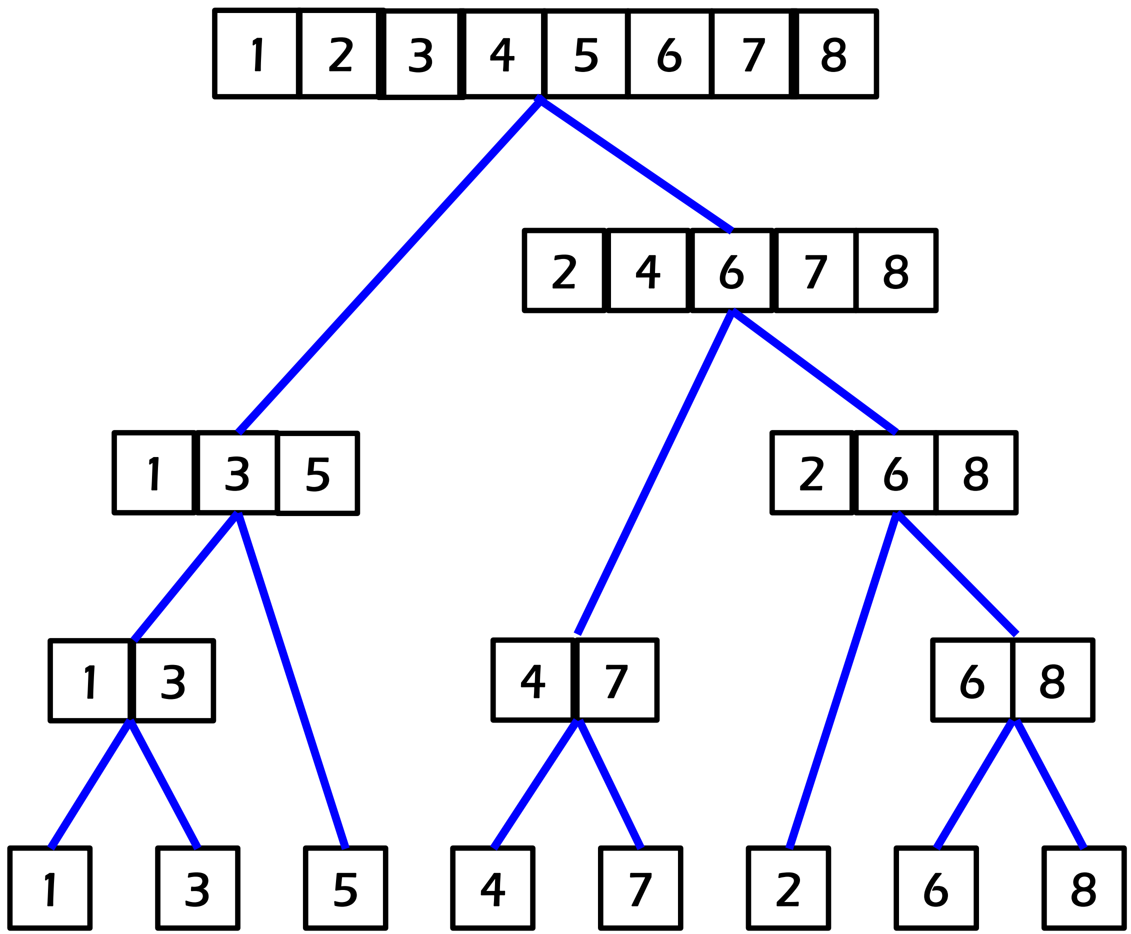
Linkage Tree Model



Linkage Tree Model



Linkage Tree Model



Linkage Tree Model

- Phép lai ghép thông thường không tương thích với mô hình Linkage Tree.

➢ Sử dụng phép biến đổi Gene-pool Optimal Mixing (GOM).

Gene-pool Optimal Mixing

- Gene-pool Optimal Mixing: Biến đổi mỗi cá thể trong quần thể thành một cá thể mới có độ thích nghi tốt hơn hoặc bằng.

Gene-pool Optimal Mixing

- Tập hợp các linkage groups trong linkage tree, bỏ đi nút gốc.

$$F = \{\{1\},\{2\},\{3\},\{4\},\{5\},\{6\},\{7\},\{8\},\\ \{1,3\},\{4,7\},\{6,8\},\\ \{1,3,5\},\{2,6,8\},\\ \{2,4,6,7,8\}\}$$

- Xáo trộn ngẫu nhiên (*shuffle*) các phần tử trong tập hợp.

Gene-pool Optimal Mixing

Cách áp dụng GOM lên một cá thể $\boldsymbol{x}$.

- Duyệt từng nhóm liên kết F^i trong F theo thứ tự ngẫu nhiên.

1. Với mỗi F^i, chọn ngẫu nhiên một cá thể $\boldsymbol{d}$ trong quần thể.
2. Sao chép các giá trị các biến thuộc F^i từ $\boldsymbol{d}$ vào $\boldsymbol{x} \rightarrow \boldsymbol{x}'$.
3. Nếu $\boldsymbol{x}'$ có độ thích nghi tốt hơn (hoặc bằng) $\boldsymbol{x}$ thì chấp nhận thao tác sao chép $\boldsymbol{x} \leftarrow \boldsymbol{x}'$.
   Ngược lại, quay trở lại giá trị $\boldsymbol{x}$ trước khi sao chép.
4. Quay lại bước 1 đến khi xét hết các nhóm liên kết trong F.

Gene-pool Optimal Mixing

$$F^i = \{2, 6, 8\}$$

0	0	0	0	0	1	1	0

Gene-pool Optimal Mixing

$$F^i = \{2, 6, 8\}$$

0	0	0	0	0	1	1	0
1	1	1	1	0	1	0	1

1	2	3	4	5	6	7	8

Gene-pool Optimal Mixing

$$F^i = \{2, 6, 8\}$$

0	0	0	0	0	1	1	0
1	1	1	1	0	1	0	1

1	2	3	4	5	6	7	8

0	1	0	0	0	1	1	1

Gene-pool Optimal Mixing

$$F^i = \{2, 6, 8\}$$

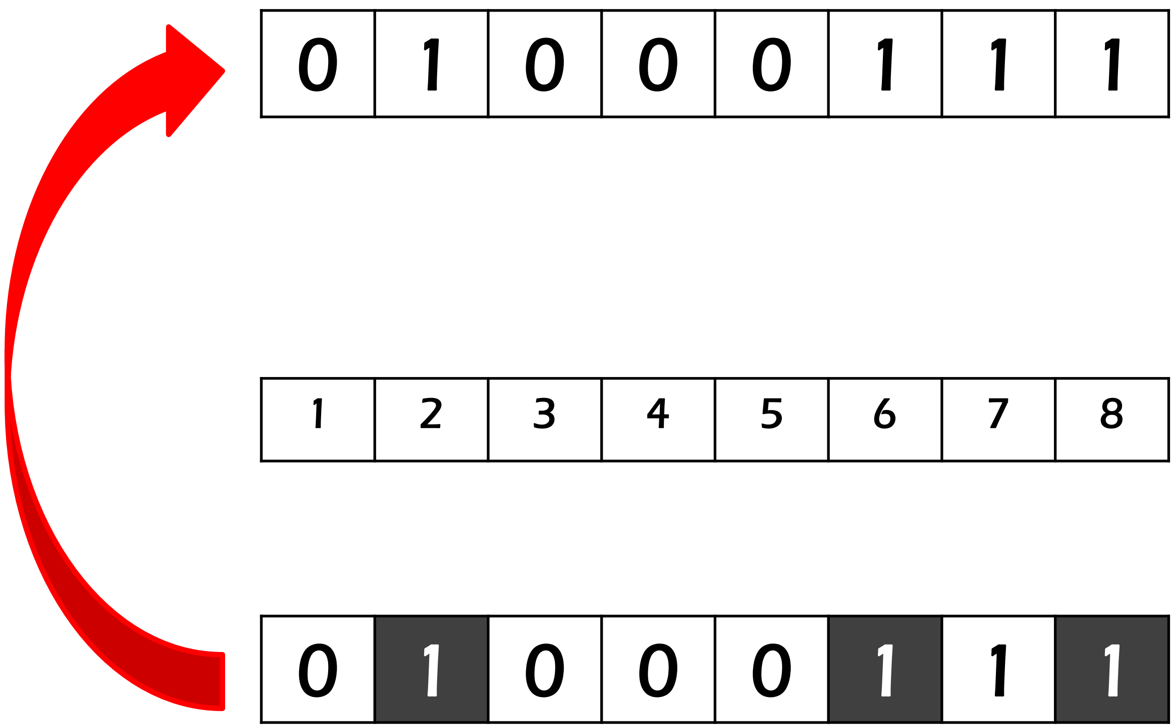
0	0	0	0	0	1	1	0
1	1	1	1	0	1	0	1

1	2	3	4	5	6	7	8

0	1	0	0	0	1	1	1

Gene-pool Optimal Mixing

$$F^i = \{2, 6, 8\}$$



Gene-pool Optimal Mixing

$$F^i = \{4, 7\}$$

0	1	0	0	0	1	1	1

Gene-pool Optimal Mixing

$$F^i = \{4, 7\}$$

0	1	0	0	0	1	1	1
0	0	1	0	0	1	0	1

1	2	3	4	5	6	7	8

Gene-pool Optimal Mixing

$$F^i = \{4, 7\}$$

0	1	0	0	0	1	1	1
0	0	1	0	0	1	0	1

1	2	3	4	5	6	7	8

0	1	0	0	0	1	0	1

UIT
TRƯỜNG ĐẠI HỌC
CÔNG NGHỆ THÔNG TIN

Gene-pool Optimal Mixing

$$F^i = \{4, 7\}$$

0	1	0	0	0	1	1	1
0	0	1	0	0	1	0	1

1	2	3	4	5	6	7	8

0	1	0	0	0	1	0	1

Gene-pool Optimal Mixing

$$F^i = \{2, 4, 6, 7, 8\}$$

0	1	0	0	0	1	1	1

Gene-pool Optimal Mixing

$$F^i = \{2, 4, 6, 7, 8\}$$

0	1	0	0	0	1	1	1
1	1	0	1	0	1	1	0

1	2	3	4	5	6	7	8

Gene-pool Optimal Mixing

$$F^i = \{2, 4, 6, 7, 8\}$$

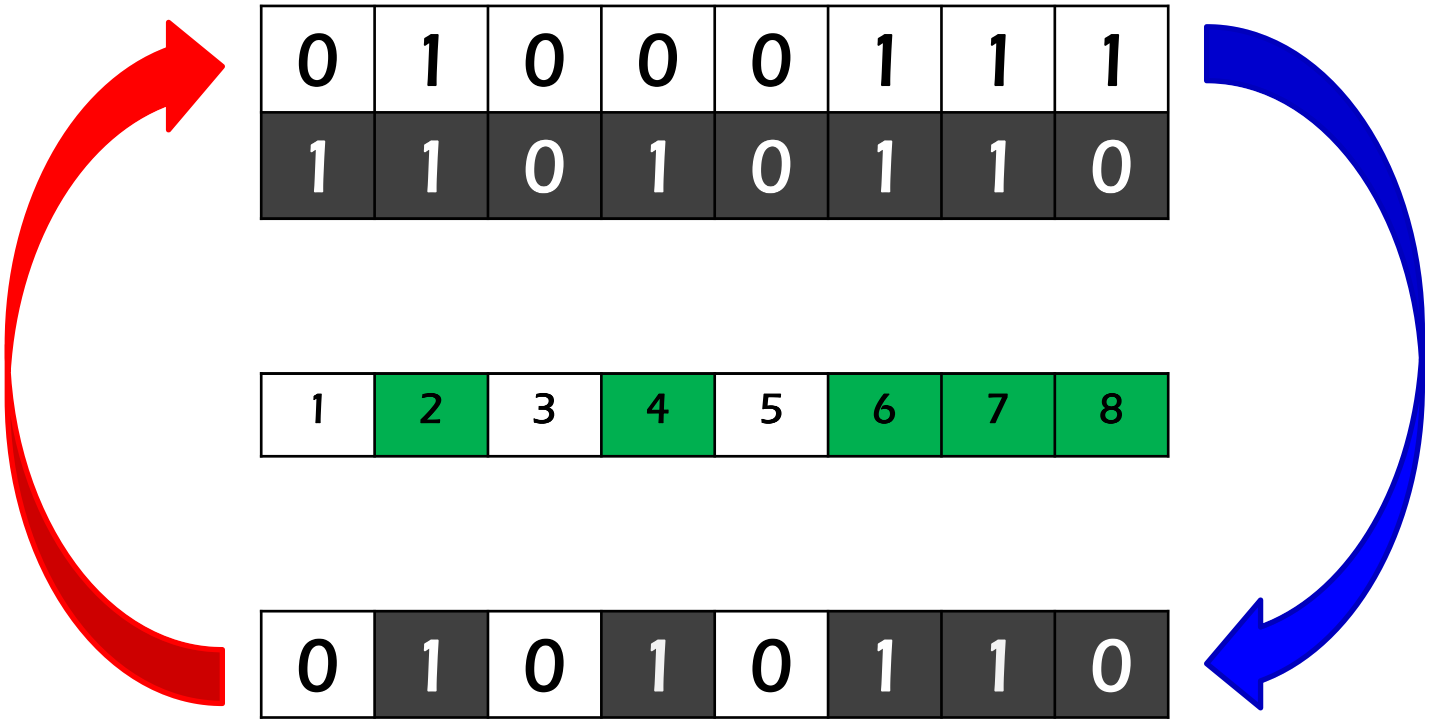
0	1	0	0	0	1	1	1
1	1	0	1	0	1	1	0

1	2	3	4	5	6	7	8

0	1	0	1	0	1	1	0

Gene-pool Optimal Mixing

$$F^i = \{2, 4, 6, 7, 8\}$$



Gene-pool Optimal Mixing

$$F^i = \cdots$$

0	1	0	1	0	1	1	0

GOMEA

- Gene-pool Optimal Mixing Evolutionary Algorithms
  GOMEA
  (Thierens & Bosman, 2011)

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

- Các thuật toán tiến hoá hoạt động hiệu quả khi nào?

➢ Khi cấu trúc của vấn đề có thể được biểu diễn đúng.

➢ Khi các phép biến đổi được thực hiện phù hợp với cấu trúc của vấn đề.